HAL open science

Multi-layer Sparse Matrix Factorization

Luc Le Magoarou, Rémi Gribonval

To cite this version:

Luc Le Magoarou, Rémi Gribonval. Multi-layer Sparse Matrix Factorization. SPARS 2015 Signal Processing with Adaptive Sparse Structured Representations, Jul 2015, Cambridge, United Kingdom. hal-01158057

HAL Id: hal-01158057
https://inria.hal.science/hal-01158057

Submitted on 29 May 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Multi-layer Sparse Matrix Factorization

Luc Le Magoarou, Rémi Gribonval
Inria Rennes-Bretagne Atlantique, Campus de Beaulieu, 35042 Rennes, France

***Abstract*—The applicability of many signal processing and data analysis techniques is limited by their prohibitive computational complexity. The cost of such techniques is often dominated by the application of large linear operators. This short paper introduces an algorithm aimed at reducing the complexity of applying such operators by approximately factorizing the corresponding matrix into few sparse factors. The proposed approach, which relies on recent advances in non-convex optimization, is first exposed, and then demonstrated experimentally.**

I. Introduction

Applying, *storing* or *estimating* a matrix $\mathbf{Y} \in \mathbb{R}^{m\times n}$ in high dimension are computationally demanding tasks. Indeed, such operations typically scale in $\mathcal{O}(mn)$, which can be prohibitive in some cases. However, there are linear operators that can be manipulated way more efficiently, such as most popular transforms (Fourier, wavelets, Hadamard, DCT...). Those special linear operators actually gain their efficiency from the fact that they can be factorized into few sparse matrices, as follows[1]:

$$\mathbf{Y} = \prod_{j=1}^{Q} \mathbf{S}_j, \tag{1}$$

where the $\mathbf{S}_j$s are sparse matrices, each having s_j non-zero entries. In that case the computational savings for the three tasks mentioned above are of the order of the Relative Complexity RC := $\sum_{j=1}^{Q} s_j/mn$ of the factorization.

This work introduces a method to compute such approximate multi-layer sparse factorizations for matrices of interest, in order to provide computational gains, and also, as a side effect, performance gains in certain applications.

II. Optimization problem

The goal being to factorize an input matrix into Q factors, with sparsity constraints on the factors, it is quite natural to propose the following optimization problem:

$$\underset{\lambda,\mathbf{S}_1,\ldots,\mathbf{S}_Q}{\text{Minimize}} \quad \frac{1}{2}\Big\|\mathbf{Y} - \lambda\prod_{j=1}^{Q}\mathbf{S}_j\Big\|_F^2 + \sum_{j=1}^{Q}\delta_{\mathcal{E}_j}(\mathbf{S}_j), \tag{2}$$

where the $\delta_{\mathcal{E}_j}$s are indicator functions of sets enforcing sparsity and normalization constraints (hence the presence of the scalar λ). Such a problem can be handled by a Proximal Alternating Linearized Minimization (PALM) algorithm [4], with convergence guarantees toward a stationary point of the objective.

III. Hierarchical factorization

In the hope of attaining better local minima, and inspired by optimization techniques used in deep learning [6], the proposed approach summarized in Algorithm 1 is a hierarchical factorization of the input matrix, consisting of a sequence of factorizations into only 2 factors of some residual. Such a decomposition of a difficult non-convex optimization problem into a sequence of smaller ones is actually reminiscent of greedy layer-wise training of deep neural networks, that has been experimentally shown to be beneficial [3]. In Algorithm 1, the factorizations of line 3 and line 5 are done using the PALM algorithm, initialized with the current values of the parameters for the global optimization step (line 5).

This work was supported in part by the European Research Council, PLEASE project (ERC-StG- 2011-277906).

[1]The product being taken from right to left: $\prod_{i=1}^{N}\mathbf{A}_i = \mathbf{A}_N\cdots\mathbf{A}_1$

Algorithm 1 Hierarchical factorization algorithm.

Input: Matrix $\mathbf{Y}$; number of factors Q; constraint sets $\mathcal{E}_k$ and $\tilde{\mathcal{E}}_k$, $k \in \{1\ldots Q-1\}$.
1: $\mathbf{T}_0 \leftarrow \mathbf{Y}$
2: **for** $k = 1$ to $Q-1$ **do**
3: Factorize the residual $\mathbf{T}_{k-1}$ into 2 factors: $\mathbf{T}_{k-1} \approx \lambda'\mathbf{F}_2\mathbf{F}_1$
4: $\mathbf{T}_k \leftarrow \lambda'\mathbf{F}_2$ and $\mathbf{S}_k \leftarrow \mathbf{F}_1$
5: Global optimization: $\mathbf{Y} \approx \lambda\mathbf{T}_k\prod_{j=1}^{k}\mathbf{S}_j$
6: **end for**
7: $\mathbf{S}_Q \leftarrow \mathbf{T}_{Q-1}$
Output: The estimated factorization: λ,$\{\mathbf{S}_j\}_{j=1}^{Q}$.

IV. Experiments

A dictionary-based image denoising experiment was performed, where a dictionary constrained to take the form of (1) is learned on 10 000 noisy 8×8 image patches (Sparse Dictionary Learning, SDL), and then used to denoise the entire image using Orthogonal Matching Pursuit (OMP) [8] with 5 dictionary atoms. The proposed method (SDL) is compared with the methods KSVD [2], ODL [7] and KSVDS [9]. Results for this experiment are summarized in Table I, where the PSNR is given at the learning stage (for the training patches) and at the denoising stage (for the entire image), as well as the relative complexity of the learned dictionary. It can be seen that the proposed approach is not only up to 8 times more computationally efficient than the others, but also leads to better denoising performances, despite poorer performances at the training stage. This can be explained by the smaller number of learned parameters leading to better generalization properties [5]. An image denoising example is also given in Figure 2.

TABLE I
Image denoising results, averaged over the standard image database taken from [1] (12 standard grey 512×512 images). The best result of each column is bold.

	Learning (PSNR)	Denoising (PSNR)	Complexity (RC)
KSVD	**24.71**	27.55	1.00
ODL	24.62	27.51	1.00
KSVDS	24.16	27.64	0.41
SDL	23.63	**29.38**	**0.13**

Another experiment was performed, where the hierarchical factorization approach was applied on an input matrix being the n-dimensional Hadamard dictionary. The method was actually able to automatically retrieve a factorization as efficient as the fast Hadamard transform ($2n\log n$ non-zero entries). This is shown in Figure 1, where the hierarchical factorization process is illustrated for $n = 32$.

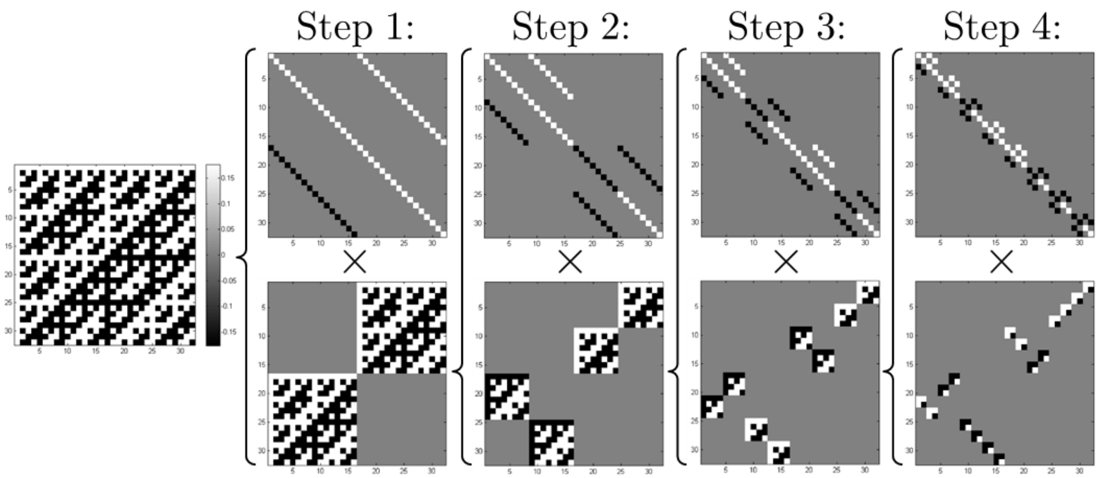

Fig. 1. Hierarchical factorization of the Hadamard matrix of size 32×32. The matrix is iteratively factorized into 2 factors, until we have $M = 5$ factors, each having $p = 64$ non-zero entries.

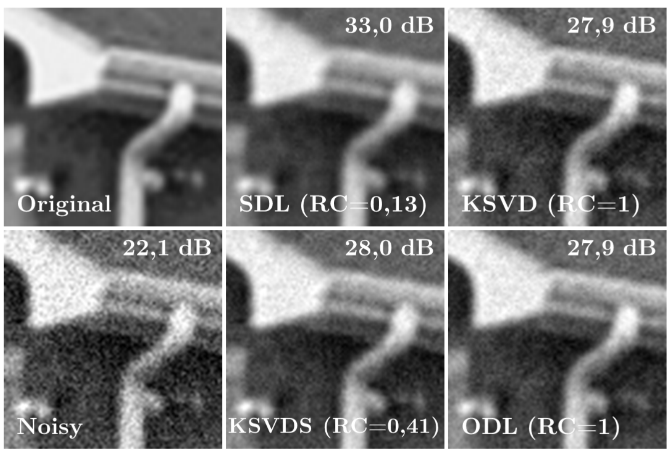

Fig. 2. Example of denoising result with Gaussian noise level $\sigma = 20$. It is a zoom on a small part of the "house" standard image. The proposed method (SDL) is compared with the methods KSVD of [2], ODL of [7] and KSVDS of [9]. Each method is presented with its relative complexity RC and SDR.

REFERENCES

[1] http://www.imageprocessingplace.com/.

[2] M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *Signal Processing, IEEE Transactions on*, 54(11):4311–4322, Nov 2006.

[3] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle. Greedy layer-wise training of deep networks. *Advances in neural information processing systems*, 19:153, 2007.

[4] J. Bolte, S. Sabach, and M. Teboulle. Proximal Alternating Linearized Minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, pages 1–36, 2013.

[5] R. Gribonval, R. Jenatton, F. Bach, M. Kleinsteuber, and M. Seibert. Sample Complexity of Dictionary Learning and other Matrix Factorizations. *ArXiv e-prints*, Dec. 2013.

[6] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[7] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11(1):19–60, Jan. 2010.

[8] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, Dec. 1993.

[9] R. Rubinstein, M. Zibulevsky, and M. Elad. Double sparsity: learning sparse dictionaries for sparse signal approximation. *IEEE Transactions on Signal Processing*, 58(3):1553–1564, Mar. 2010.